

Control Inteligente

Diseño de Controladores Fuzzy

Luis Moreno, Santiago Garrido, Dorin Copaci

Dpto. Ing. de Sistemas y Automática
Universidad Carlos III
Madrid

Oct 2019



Table of contents

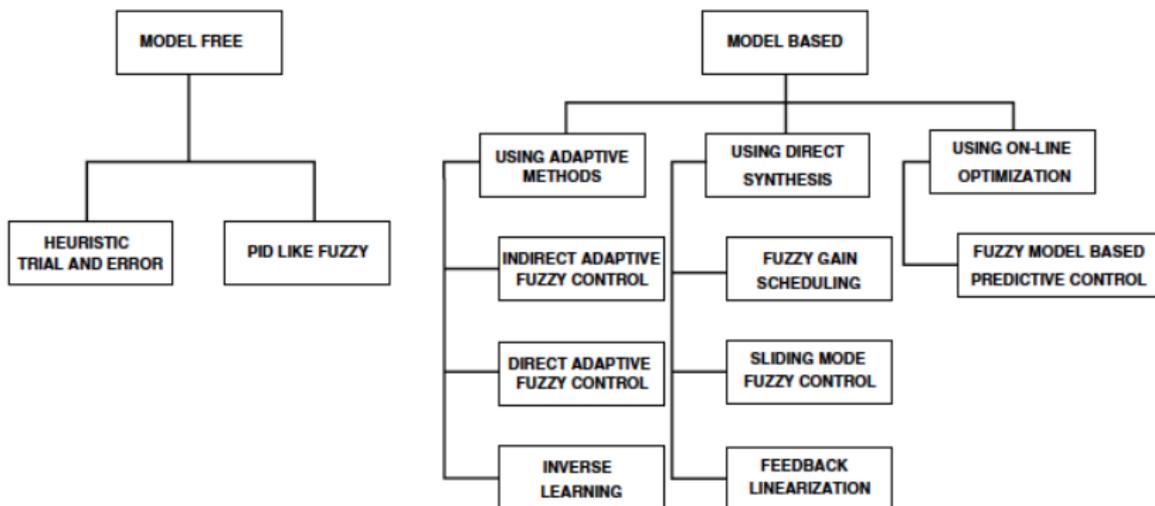
- 1 Control borroso.
- 2 Controladores borrosos sin modelo
- 3 Controladores borrosos basados en modelo

Introducción al control borroso

- De acuerdo con la información usada en su diseño podemos distinguir dos tipos de controladores fuzzy o borrosos:
 - **Controladores sin modelo.** Estas estrategias no se basan en modelos matemáticos completos, aunque no son completamente libres de modelo ya que se basan en la información extraída de experimentos simples o a partir de un modelo heurístico presente en la mente del experto.
 - **Controladores basados en modelo.** Estos controladores usualmente requieren una descripción completa de la dinámica de la planta.

Clasificación de los controladores fuzzy

Controladores fuzzy.



Controladores borrosos sin modelo

Heurísticos

- Esta metodología es la primera que se usó en el diseño de controladores borrosos y se basa en la experiencia acumulada a lo largo de años de control manual de los sistemas.
- El enfoque típico para construir los controladores consiste en:
 - Formular la base de reglas usando la información suministrada por el operador que controla manualmente la planta.
 - La mayoría de las veces estos controladores se usan a alto nivel como un tipo de **control supervisor** y en el aspectos como la estabilidad no son críticos.
 - La estabilización de la planta es una tarea que realizan los controladores de bajo nivel.
 - En muchos casos el controlador borroso no se usa directamente en modo automático sobre el sistema, sino como un sistema de ayuda al operador.
 - Ejemplos: control de hornos de cemento, boiler startup sequences, washing powder production, incineración de residuos y tratamiento de aguas residuales.

Controladores borrosos sin modelo

Tipo PID

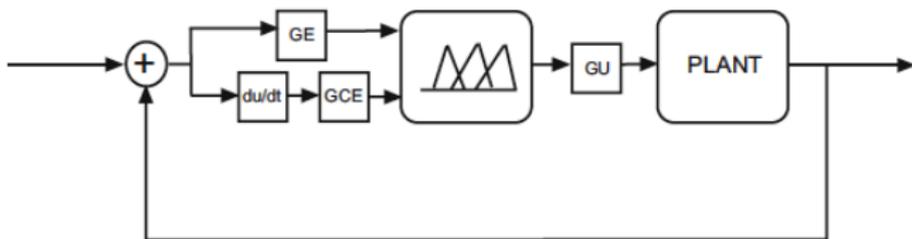
- Estos controladores borrosos tipo PID (proporcional integral derivativo) pueden ser diseñados usando los mismos experimentos que se usan para ajustar controladores lineales PID sin modelo o usando modelos simples de respuesta (respuesta a un escalón).
- La idea básica de este enfoque es que cualquier PID con entrada y salida acotada puede ser reproducido con exactitud por un sistema borroso.
- El método de diseño consiste en:
 - ① Ajustar un controlador PID controller usando uno de los métodos tradicionales de ajuste (Ziegler Nichols o Kappa-Tau de Aström).
 - ② Construir un controlador borroso equivalente al PID ajustado.
 - ③ Hacer ajustes posteriores sobre el controlador borroso usando heurísticos.

- El caso más frecuente de estos controladores borrosos de bajo nivel (nosupervisor) son los controladores del tipo: $e, \Delta e$.
- Estos controladores suelen tener dos entradas: e =error and Δe =cambio error, y una salida, que suele ser la acción de control u o Δu dependiendo de si el controlador actúa como PD (proporcional derivativo) o PI (proporcional integral).
- Una de sus características es que se puede establecer una analogía directa con los controladores lineales clásicos PD y PI.

Controladores borrosos tipo PID

PD

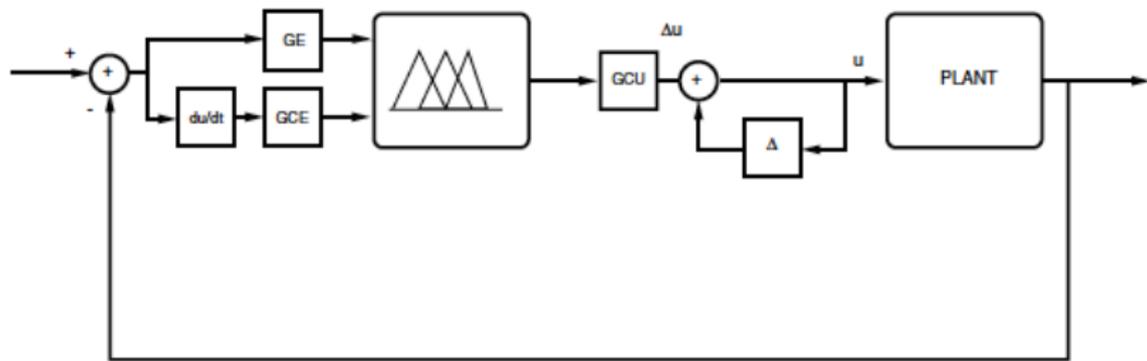
Estructura de un controlador PD borroso .



Controladores borrosos tipo PID

PI

Estructura de un controlador PI borroso .



Controladores borrosos tipo PID

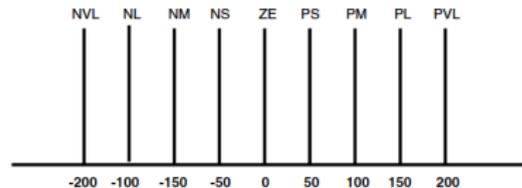
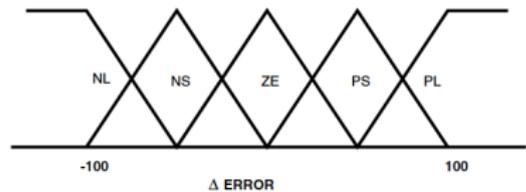
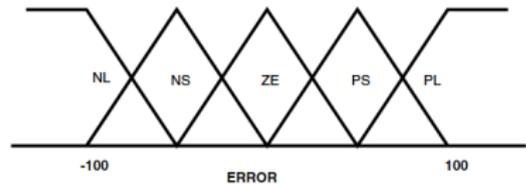
- Asumiendo cinco conjuntos borrosos (NL: Negative Large, NS: Negative Small, ZE: Zero, PS: Positive Small and PL: Positive Large) para cada entrada
- y nueve conjuntos borrosos para la salida (los cinco previos más PM: Positive Medium, NM: Negative medium, PVL: Positive Very Large and NVL: Negative Very Large)
- una base de reglas típica es de la forma:

$e \setminus \Delta e$	NL	NS	ZE	PS	PL
NL	PVL	PL	PM	PS	ZE
NS	PL	PM	PS	ZE	NS
ZE	PM	PS	ZE	NS	NM
PS	PS	ZE	NS	NM	NL
PL	ZE	NS	NM	NL	NVL

Controladores borrosos tipo PID

Observaciones

- Notas
 - La distribución de las f. de pertenencia de las entradas puede ser muy uniforme.
 - Las f. de pertenencia de la salida han sido escaladas entre -200 y 200 y usando singletons.
 - Con esta base de reglas, y f. de pertenencia triangulares igualmente distribuidas con solapamiento en 0.5, se puede obtener un contr. fuzzy equivalente al PI o PD.



Controladores borrosos tipo PID

- Esta característica es un mecanismo muy seguro para ajustar un controlador borroso ya que reemplazamos la PD por su versión borrosa equivalente para un posterior ajuste más fino de las f de pertenencia para mejorarlo.
- Método:
 - Para reemplazar un controlador PD descrito por la función de transferencia

$$C(s) = K_p(1 + \tau_d s) \quad (1)$$

los factores de escala deben cumplir lo siguiente:

- $GE \times GU = K_p$
- $GCE \times GU = K_p \tau_d$
- $GE \times \max |e| \leq 100$
- $GCE \times \max |\Delta e| \leq 100$

Controladores borrosos tipo PID

- Método:
 - Para reemplazar un controlador PI descrito por la función de transferencia

$$C(s) = K_p \left(1 + \frac{1}{\tau_i s} \right) \quad (2)$$

los factores de escala deben cumplir lo siguiente:

- $GCE \times GCU = K_p$
 - $GE \times GCU = \frac{K_p}{\tau_i}$
 - $GE \times \text{máx}|e| \leq 100$
 - $GCE \times \text{máx}|\Delta e| \leq 100$
- Obsérvese que en ambos casos las dos primeras condiciones garantizan el mapeo correcto de las ganancias de controlador y las dos últimas sirven para evitar la saturación.

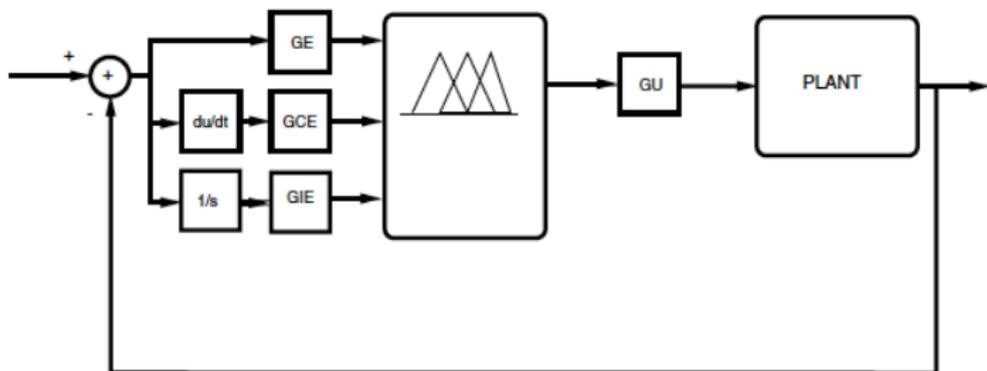
Controladores borrosos tipo PID

Controlador PID

- El diseño de un controlador borroso PID puede hacerse de varias formas:
 - Diseñando un controlador con tres entradas: el error (acción proporcional), el delta del error (acción derivativa) y la suma de los errores (acción integral) (ver contr. borroso PID). El inconveniente de este controlador es que **crece mucho el número de reglas** y en vez de las 25 que teníamos en los controladores vistos anteriormente (para un sistema con 5 f. de pertenencia en las entradas) pasamos a un controlador con 125 reglas lo que hace el ajuste mucho más difícil.
 - Un diseño más eficiente consiste en dividir el controlador en dos controladores, uno es el controlador equivalente PD y otro que suministra la acción integral. Esto **reduce considerablemente el número de reglas** (en este caso a 30) (ver contr. borroso PD+I).

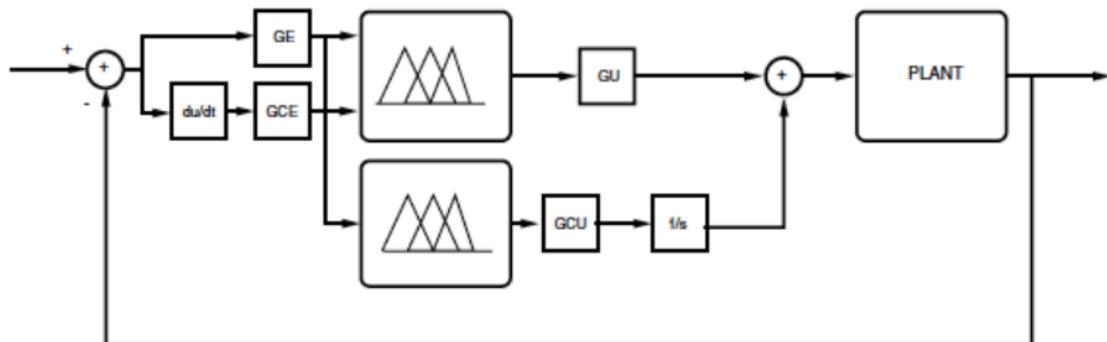
Controladores borrosos tipo PID

Controlador PID borroso .



Controladores borrosos tipo PID

Controlador PD+I borroso .



Controladores borrosos tipo PID

Mejoras

- El uso de constantes de ganancia puede evitarse usando mecanismos de inicialización del sistema borroso más complejos donde las entradas no estén escaladas y los singletons de la salida se calculen usando su relación directa con los valores de las f. de pertenencia de las entradas, también se pueden inicializar la reglas .
- Otras mejoras pueden conseguirse mediante **prueba y error**, si *no hay modelo* de la planta.
- O bien mediante **métodos de optimización**, si se tiene un *modelo disponible*.

Controladores borrosos basados en modelo

Model Based Fuzzy Control

- Veremos tres tipos de controladores borrosos basados en modelo.
- Estos son los más representativos pero en la práctica pueden aparecer mezclados hasta un cierto punto.
 - **Métodos adaptivos.** Se basa en el modelo de la planta y simulaciones para optimizar los parámetros del controlador borroso. En la mayoría de los casos estas simulaciones tienen que realizarse *offline*, pero en los métodos de control adaptativo directo la optimización puede ser realizada online.
 - **Métodos de síntesis directa.** Esta técnica usa o la información dada por los parámetros del modelo o el modelo mismo para construir el controlador.
 - **Métodos de optimización online.** Esta técnica usa un modelo borroso para predecir el comportamiento futuro de la planta en un cierto horizonte temporal (receding horizon) y con esta información se calculan los movimientos futuros de las acciones de control usando métodos de optimización.

Métodos adaptativos

Aprendizaje inverso (Inverse learning)

- La idea fundamental de este tipo de controladores es la construcción de un **modelo inverso de la planta** tal que el controlador genere una entrada que conduzca el estado de la planta al sistema desde el estado actual x_k hasta el estado deseado x_{k+n}^d .
 - Para que este método sea aplicable se asume que los estados de la planta son *medibles*.
 - Se asume que la dinámica de la planta es discreta o al menos muestreada y representada por la función

$$x_{k+1} = f(x_k, u_k) \quad (3)$$

donde k representa el tiempo discreto, x_k es el estado y u_k es la entrada a la planta.

- El estado de la planta para el instante de tiempo $k + N$ viene dado por

$$x_{k+N} = \underbrace{f(f(f(x_k, u_k), u_{k+1}), \dots, u_{k+N-1})}_{N \text{ times}} \quad (4)$$

Métodos adaptativos

Aprendizaje inverso (Inverse learning)

- Esto es equivalente a decir que:

$$x_{k+N} = F(x_k, U) \quad (5)$$

donde F es una función que representa la composición múltiple de la función $f(., .)$ y U es un vector con la secuencia de entrada u_k, \dots, u_{k+N-1} .

- Con esta descripción y asumiendo la **invertibilidad** de la función F , puede construirse un mapa inverso de la planta tal que

$$U = G(x_k, x_{k+N}) \quad (6)$$

- Esta función generará la secuencia de control U necesaria para mover la planta desde el estado actual x_k hasta el estado x_{k+N} en N pasos.
- La existencia de este mapa inverso es equivalente a la condición de controlabilidad en los sistemas lineales.

Métodos adaptativos

Aprendizaje inverso (Inverse learning)

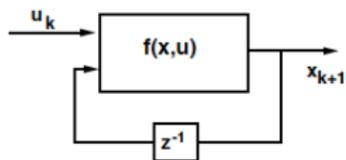
- La existencia del mapa G no garantiza la existencia de una forma analítica cerrada.
- El sistema borroso \hat{G} se usa para aproximar este mapa.
- La dimensión del mapa para un sistema de orden n con una entrada tendrá $2n$ entradas y N salidas.
- Resulta evidente que incluso para sistemas de orden bajo el número de entradas hará que el sistema sea muy grande (con un gran número de reglas).
- El uso de modelos inversos de control opera del siguiente modo:
 - La referencia se da como el estado futuro x_{k+N}^d y junto con el estado actual el modelo inverso \hat{G} genera un vector U .
 - La primera entrada del vector se implementa y la función es evaluada una vez más en el siguiente tiempo de muestreo.
 - Si la referencia no es conocida con anticipación, la referencia futura será reemplazada por la referencia actual, generándose un sistema que se comporta como un retardo puro.
 - Por razones prácticas es preferible usar un modelo inverso que genere sólo un valor para la secuencia de entrada con $N = 1$.

Métodos adaptativos

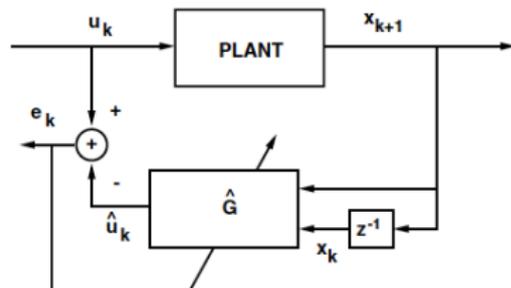
Aprendizaje inverso (Inverse learning)

- Esquema de diseño de un controlador inverso.

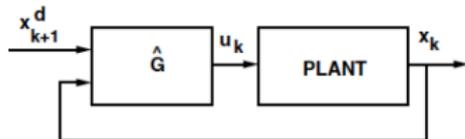
- a Planta
- b Fase de aprendizaje
- c Esquema de operación



(a)



(b)



(c)

Métodos adaptativos

Aprendizaje inverso (Inverse learning)

- Limitaciones
 - Está limitada por la **condición de invertibilidad** de la planta y el hecho de que la minimización de la norma $\|u_k - \hat{u}_k\|$ no garantiza la minimización de $\|x_k - x_k^d\|$.
 - Su uso es muy limitado ya que demanda un acceso completo a los estados del sistema.
 - El ajuste (tuning) requiere que un gran número de posibles transiciones desde x_k a x_{k+1} sean comprobadas para garantizar la cobertura completa del rango de operación del controlador.
 - Este problema es peor cuando el orden del sistema es grande. Para un sistema de 5 orden ya se pueden generar algunos serios problemas, porque el controlador tendrá $2 \times 5 = 10$ entradas y al menos 1024 reglas.
 - Sólo se puede usar con sistemas de orden bajo (1 o 2 orden), invertibles y con acceso completo a la variable de estado del sistema.
 - Las prestaciones están limitadas por los experimentos usados durante la fase de entrenamiento (una especie de fase de identificación)
 - Algunos esquemas propuestos actualizan de forma continua y online el modelo inverso mientras el proceso está en operación.

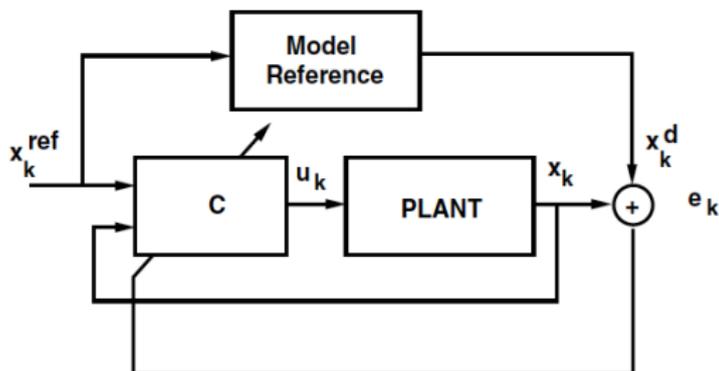
Métodos adaptativos

Control borroso adaptivo directo

- Esta técnica ajusta los parámetros del controlador de acuerdo a alguna medida de rendimiento del sistema.
- En este caso el controlador está representado por un sistema borroso.
- Existen muchos esquemas posibles de este tipo pero se verán dos casos: la **realimentación de estado** y la **realimentación de salida**.
- La medida de las prestaciones del sistema puede usar alguna de las especificaciones (tiempo de establecimiento, sobreoscilación máxima, tiempo de subida) o un modelo de referencia (normalmente uno lineal) que ya incluya las especificaciones deseadas.

Control borroso adaptivo directo

Controlador borroso adaptivo directo.



Control borroso adaptivo directo

Realimentación de estado

- Dado el modelo de una planta

$$x_{k+1} = f(x_k, u_k) \quad (7)$$

el objetivo consiste en diseñar un sistema de control

$$u_k = g(x_k, x_k^{ref}, \theta) \quad (8)$$

- Este controlador es un *mapa estático no lineal* con $2n$ entradas y una parametrización dada por el vector θ . El mapa no lineal puede ser construido mediante un sistema borroso.
- El objetivo es obtener un conjunto de parámetros θ tal que la dinámica en bucle cerrado debe replicar la dinámica del *modelo de referencia* mostrado en la siguiente ecuación.

$$x_{k+1}^d = \bar{f}(x_k^d, x_k^{ref}) \quad (9)$$

donde x_k^d es la salida del modelo de referencia y x_k^{ref} es la referencia.

Control borroso adaptivo directo

Realimentación de estado

- El diseño de este controlador puede formularse por medio del problema de optimización

$$\min_{\theta} J = \min_{\theta} \sum_{k=1}^N \|x_k^d - x_k\| \quad (10)$$

la expresión en bucle cerrado del sistema es

$$x_{k+1} = f(x_k, g(x_k, x_k^{ref}, \theta)) \quad (11)$$

entonces la minimización puede escribirse como

$$\min_{\theta} \sum_{k=0}^{N-1} \|\bar{f}(x_k^d, x_k^{ref}) - f(x_k, g(x_k, x_k^{ref}, \theta))\|^2 \quad (12)$$

Control borroso adaptivo directo

Realimentación de estado

- Si se va a aplicar un descenso del gradiente en la optimización, los ajustes serán dirigidos por el gradiente

$$\frac{\partial J}{\partial \theta} = 2 \sum_{k=0}^{N-1} -(x_{k+1}^d - x_{k+1}) \frac{\partial x_{k+1}}{\partial \theta} \quad (13)$$

donde el gradiente $\frac{\partial x_{k+1}}{\partial \theta}$ es generado por el sistema dinámico

$$\begin{aligned} \frac{\partial x_{k+1}}{\partial \theta} = & \left[\frac{\partial f(x_k, u_k)}{\partial x_k} + \frac{\partial f(x_k, u_k)}{\partial u_k} \frac{\partial g(x_k, x_k^{ref})}{\partial x_k} \right] \frac{\partial x_k}{\partial \theta} \\ & + \frac{\partial f(x_k, u_k)}{\partial u_k} \frac{\partial g(x_k, x_k^{ref})}{\partial \theta} \end{aligned} \quad (14)$$

Control borroso adaptivo directo

Realimentación de estado

- Está claro que para poder aplicar este método se requiere un buen modelo de la planta $f(., .)$.
- Ya que es necesario derivarlas expresiones $\frac{\partial f(x_k, u_k)}{\partial x_k}$ y $\frac{\partial f(x_k, u_k)}{\partial u_k}$, bien de forma analítica o por medio de métodos numéricos.

Control borroso adaptivo directo

Realimentación de la salida

- Asumiendo que el modelo de la planta puede describirse de la forma

$$\begin{aligned}x_{k+1} &= f(x_k, u_k) \\ y_k &= g(x_k)\end{aligned}\tag{15}$$

y el controlador será un modelo borroso dinámico representado por la ecuación:

$$\begin{aligned}z_{k+1} &= h(z_k, y_k, y_k^{ref} | \alpha) \\ u_k &= d(z_k, y_k, y_k^{ref} | \beta)\end{aligned}\tag{16}$$

donde:

- z_k son los estados del controlador
- y_k^{ref} es la entrada de referencia
- α y β son los parámetros de las funciones $h(., ., .)$ y $d(., ., .)$.

Control borroso adaptivo directo

Realimentación de la salida

- El sistema en cadena cerrada puede ser expresado como el sistema dinámico:

$$\begin{aligned} s_{k+1} &= m(s_k, y_k^{ref} | \theta) \\ u_k &= p(s_k, y_k^{ref} | \beta) \\ y_k &= q(s_k) \end{aligned} \tag{17}$$

donde:

- $s_k = x_k, z_k^T$,
- $\theta = \alpha, \beta^T$,
- $d(z_k, y_k, y_k^{ref} | \beta) = p(s_k, y_k^{ref} | \beta)$
- $y \ q(s_k) = g(x_k)$.

Control borroso adaptivo directo

Realimentación de la salida

- Asumiendo una función de coste de la forma

$$\min_{\theta} J = \min_{\theta} \sum_{k=1}^N \|y_k^{ref} - y_k\| + \lambda u_k^T u_k \quad (18)$$

- la derivada vendrá dada por

$$\frac{\partial J}{\partial \theta} = 2 \sum_{k=0}^{N-1} -(y_k^{ref} - y_k) \frac{\partial y_k}{\partial \theta} + \lambda u_k \frac{\partial u_k}{\partial \theta} \quad (19)$$

con los términos $\frac{\partial y_k}{\partial \theta}$ y $\frac{\partial u_k}{\partial \theta}$ generados por el sistema dinámico:

$$\begin{aligned} \frac{\partial s_{k+1}}{\partial \theta} &= \frac{\partial m}{\partial s_k} \frac{\partial s_k}{\partial \theta} + \frac{\partial m}{\partial \theta} \\ \frac{\partial y_k}{\partial \theta} &= \frac{\partial q}{\partial s_k} \frac{\partial s_k}{\partial \theta} \\ \frac{\partial u_k}{\partial \theta} &= \frac{\partial p}{\partial s_k} \frac{\partial s_k}{\partial \theta} + \frac{\partial p}{\partial \theta} \end{aligned} \quad (20)$$

Control borroso adaptivo directo

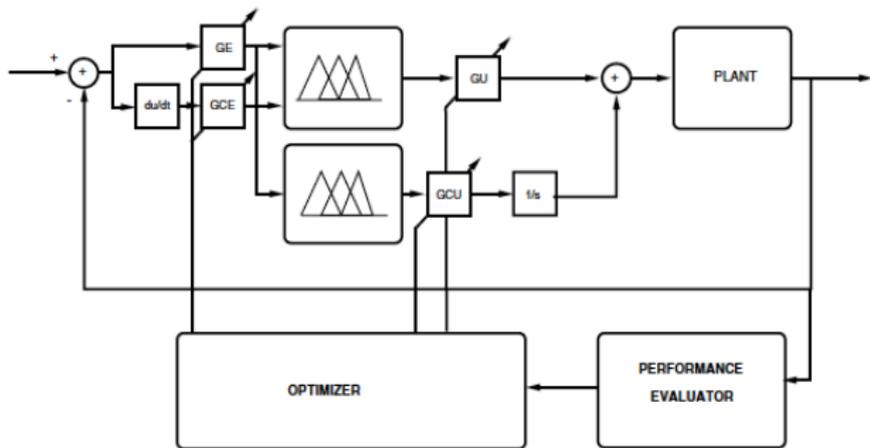
Realimentación de la salida

- Según los parámetros a ajustar (funciones de pertenencia, consecuentes o valores de escala) el cálculo de los gradientes será más o menos complejo.
- El procedimiento de optimización debe evaluar una función de coste que pueda incluir la optimización del rechazo a perturbaciones o el tracking de setpoints o un compromiso entre estos dos objetivos.
- Diferentes parámetros pueden ser ajustados usando este método, pero de acuerdo a su impacto en las prestaciones la prioridad será:
 - Ajustar los factores de escala si es que hay alguno,
 - posteriormente ajustar los consecuentes de las reglas del sistema borroso
 - y finalmente si se necesita un ajuste muy fino los parámetros de las funciones de pertenencia de los antecedentes de las reglas.
- Este método no necesita tener acceso completo a los estados y la estrategia de realimentación de la salida puede implementarse si el sistema es a la vez controlable y observable.

Control borroso adaptivo directo

Ajuste de escalas

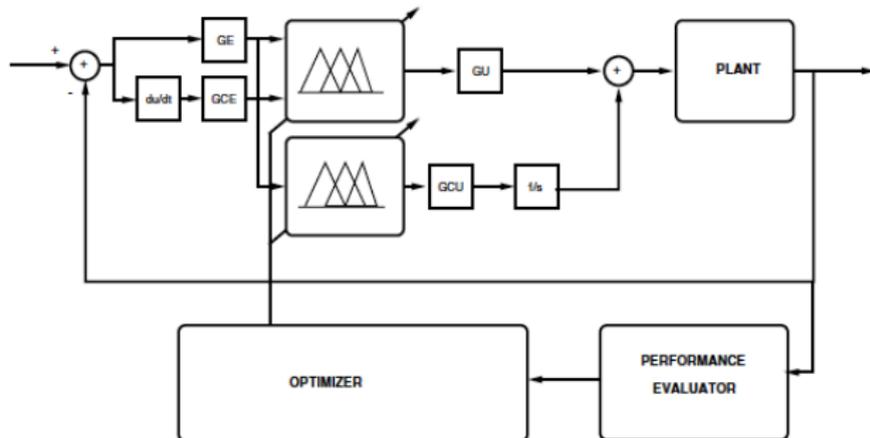
Ajuste de las constantes de ganancia usando optimización no lineal



Control borroso adaptivo directo

Ajuste de los parámetros internos

Ajuste de los parámetros internos del controlador borroso (funciones de pertenencia, consecuentes, etc.)



Síntesis directa

Linealización por realimentación

- Esta metodología puede aplicarse a sistemas no lineales de orden n de la forma

$$\begin{aligned} \dot{x}(n) &= f(x, \dot{x}, \dots, x^{(n-1)}) + g(x, \dot{x}, \dots, x^{(n-1)})u \\ y &= x \end{aligned} \quad (21)$$

- Una característica de este tipo de sistemas no lineales es que el conocimiento de las funciones $f(x)$ y $g(x)$ puede ser usado para diseñar una señal de control u tal que cancele la no linealidad y el controlador pueda diseñarse usando técnicas lineales como el posicionamiento de polos.
- La ley de control puede describirse por:

$$u = \frac{1}{g(x)}(-f(x) + y_{ref}^{(n)} + \mathbf{k}^T \mathbf{e}) \quad (22)$$

Síntesis directa

Linealización por realimentación

- Una vez que se aplica a la planta, el error de control se define como $e = y_{ref} - y$, y el vector de errores de estado se define como $e = \{e, \dot{e}, \dots, e^{(n-1)}\}^T$.
- El vector $\mathbf{k} = \{k_n, \dots, k_1\}^T$ define la dinámica del error.
- Introduciendo la ley de control en la planta se obtendrá una dinámica en bucle cerrado definida por

$$\begin{aligned}x^{(n)} &= y_{ref}^{(n)} + \mathbf{k}^T \mathbf{e} \\ y &= x\end{aligned}\tag{23}$$

y la dinámica del error será

$$e^{(n)} + k_1 e^{(n-1)} + \dots + k_n e = 0\tag{24}$$

- Se puede ver que la dinámica en bucle cerrado estará gobernada por la dinámica determinada por los componentes del vector \mathbf{k} dado que las entradas del vector son los coeficientes del polinomio característico del sistema en cadena cerrada.
- Una elección adecuada de los elementos de \mathbf{k} garantizan la estabilidad y la convergencia de y hacia y_{ref} .

Síntesis directa

Linealización por realimentación

- Los sistemas borrosos pueden usarse para representar las funciones f y g , pero existirán algunos errores de modelado.
 - La función f se aproximará por el sistema borroso descrito por la función

$$\hat{f}(x|\theta_f) = \theta_f^T \sigma(x) \quad (25)$$

donde θ_f son los consecuentes de las reglas y $\sigma(x)$ representa el proceso de inferencia.

- La función g se aproximará por

$$\hat{g}(x|\theta_g) = \theta_g^T \eta(x) \quad (26)$$

donde θ_g son los consecuentes de las reglas y $\eta(x)$ representa el proceso de inferencia.

Síntesis directa

Linealización por realimentación

- Si se tiene en cuenta la diferencia entre la función y su aproximación,
 - la dinámica del error vendrá expresada por

$$e^{(n)} = -\mathbf{k}^T \mathbf{e} + [\hat{f}(x|\theta_f) - f(x)] + [\hat{g}(x|\theta_g) - g(x)]u^* \quad (27)$$

donde

$$u^* = \frac{1}{\hat{g}(x|\theta_g)} (-\hat{f}(x|\theta_f) + y_{ref}^{(n)} + \mathbf{k}^T \mathbf{e}) \quad (28)$$

- y esta dinámica del error puede escribirse en forma matricial en la forma

$$\Lambda = \begin{pmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ & \vdots & & & \ddots & \vdots & \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ -k_n & -k_{n-1} & & & \dots & & -k_1 \end{pmatrix}, b = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \quad (29)$$

Síntesis directa

Linealización por realimentación

- Y de aquí podemos generar la ec diferencial que describe la dinámica del error

$$\dot{\mathbf{e}} = \Lambda \mathbf{e} + \mathbf{b}[\hat{f}(x|\theta_f) - f(x)] + \mathbf{b}[\hat{g}(x|\theta_g) - g(x)]u^* \quad (30)$$

- El error de aproximación mínimo se define como

$$\omega = [\hat{f}(x|\theta_f^*) - f(x)] + [\hat{g}(x|\theta_g^*) - g(x)]u^* \quad (31)$$

donde θ_f^* y θ_g^* representan los valores óptimos de las aproximaciones de θ_f y θ_g .

- Usando esta representación la dinámica del error puede reescribirse como,

$$\dot{\mathbf{e}} = \Lambda \mathbf{e} + \mathbf{b}\{[\hat{f}(x|\theta_f) - \hat{f}(x|\theta_f^*)] + [\hat{g}(x|\theta_g) - \hat{g}(x|\theta_g^*)] + \omega\} \quad (32)$$

Síntesis directa

Linealización por realimentación

- Reemplazando en las expresiones por $\hat{f}(x|\theta_f)$ and $\hat{g}(x|\theta_g)$ obtenidas anteriormente la dinámica del error queda en la forma

$$\dot{\mathbf{e}} = \Lambda \mathbf{e} + \mathbf{b}[(\theta_f - \theta_f^*)^T \sigma(\mathbf{x}) + (\theta_g - \theta_g^*)^T \eta(\mathbf{x})u^* + \omega] \quad (33)$$

- El objetivo es definir una ley de adaptación estable para θ_f and θ_g de forma que los errores de seguimiento y de los parámetros sean mínimos a lo largo de la evolución en el tiempo. Wang propuso un procedimiento basado en la función de Liapunov :

$$V = \frac{1}{2} \mathbf{e}^T P \mathbf{e} + \frac{1}{2\gamma_1} (\theta_f - \theta_f^*)^T (\theta_f - \theta_f^*) + \frac{1}{2\gamma_2} (\theta_g - \theta_g^*)^T (\theta_g - \theta_g^*) \quad (34)$$

donde γ_1 y γ_2 son constantes positivas y P es una matriz definida positiva que satisface la ecuación de Liapunov:

$$\Lambda^T P + P \Lambda = -Q \quad (35)$$

y Q es una matriz arbitraria definida positiva.

Síntesis directa

Linealización por realimentación

- La derivada de la función de Liapunov viene dada por

$$\begin{aligned}\dot{V} = & -\frac{1}{2}\mathbf{e}^T P \mathbf{e} + \mathbf{e}^T P \mathbf{b} \omega \\ & + \frac{1}{\gamma_1}(\theta_f - \theta_f^*)^T [\dot{\theta}_f + \gamma_1 \mathbf{e}^T P \mathbf{b} \sigma(\mathbf{x})] \\ & + \frac{1}{\gamma_2}(\theta_g - \theta_g^*)^T [\dot{\theta}_g + \gamma_2 \mathbf{e}^T P \mathbf{b} \eta(\mathbf{x}) u^*] \quad (36)\end{aligned}$$

- La minimización del error de tracking \mathbf{e} y de los parámetros de error es equivalente a la minimización de la función de Lyapunov V . La ley de adaptación debe garantizar que \dot{V} es negativa.

Síntesis directa

Linealización por realimentación

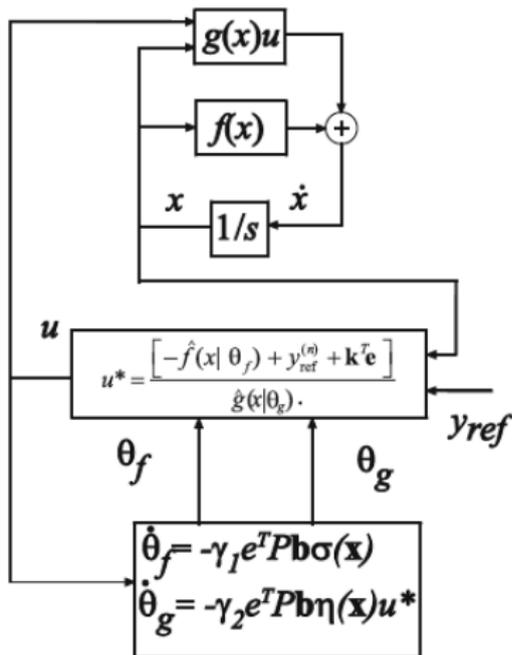
- Para garantizarlo observemos la expresión:
 - el término $-\frac{1}{2}\mathbf{e}^T P \mathbf{e}$ es siempre negativo dado que P es siempre positiva definida,
 - el término $\mathbf{e}^T P \mathbf{b} \omega$ puede ser positivo o negativo, pero si el modelado inicial se realiza correctamente ω debe ser pequeño y por tanto $\mathbf{e}^T P \mathbf{b} \omega \ll \frac{1}{2}\mathbf{e}^T P \mathbf{e}$, lo que garantiza un valor negativo para esta parte de \dot{V}
 - Finalmente, forzando a que los dos últimos términos sean iguales a cero, obtenemos las siguientes leyes de adaptación :

$$\begin{aligned}\dot{\theta}_f &= -\gamma_1 \mathbf{e}^T P \mathbf{b} \sigma(\mathbf{x}) \\ \dot{\theta}_g &= -\gamma_2 \mathbf{e}^T P \mathbf{b} \eta(\mathbf{x}) u^*\end{aligned}\tag{37}$$

Síntesis directa

Linealización por realimentación

Esquema de control para la linealización por realimentación adaptativa.



Síntesis directa

Linealización por realimentación

- Resumen de los pasos:
 - ① Diseño de un modelo inicial de la planta $\hat{f}(x|\theta_f)$, $\hat{g}(x|\theta_g)$ usando alguna técnica de identificación de las ya conocidas.
 - ② Diseño del vector \mathbf{k} de acuerdo a la conducta deseada.
 - ③ Conectar el sistema en bucle cerrado y utilizar la regla adaptativa para adaptar online los valores de los consecuentes θ_f , θ_g de las reglas.

Síntesis directa

Linealización por realimentación

- Ventajas e inconvenientes.
 - La principal ventaja de esta estrategia de control es que permite construir un controlador directamente a partir del modelo y de la conducta lineal deseada para el sistema.
 - El inconveniente principal es que sólo es aplicable a un conjunto limitado de sistemas no-lineales de tiempo continuo y adó que el controlador se implementa en tiempo discreto debe de tenerse cuidado en la fase de implementación.
 - Este esquema no supone que existan perturbaciones y se basa en que la adaptación de los parámetros de θ_f y θ_g compensan el efecto de las perturbaciones.
 - La estabilidad del sistema está garantizada, pero el error de seguimiento puede ser proporcional a la diferencia entre la planta y el modelo.

Síntesis directa

Gain Scheduling (Planificación de la ganancia)

- Esta técnica se ha usado mucho en aplicaciones industriales y control de aviones.
- La gran ventaja que presenta es que usa técnicas lineales para diseñar controladores para sistemas no-lineales mediante la aplicación de linealización alrededor de diferentes puntos de operación.
- De este modo se diseñan varios controladores lineales que se conectan a la planta de acuerdo al punto de operación en el que se encuentra en cada instante.
- Los sistemas Fuzzy ofrecen a esta técnica la posibilidad de integrar en un sólo mecanismo la detección del punto de operación y la interpolación entre los diferentes puntos de operación, generando una ley de control no lineal suave.

Síntesis directa

Gain Scheduling (Planificación de la ganancia)

- En esta técnica se suele usar un modelo de planta basado en un sistema fuzzy del tipo Takagi-Sugeno con reglas de la forma:

$$R_i : \text{ Si } x_1(k) \text{ es } \mu_{i1} \text{ y } \dots \text{ y } x_n(k) \text{ es } \mu_{in} \\ \text{ entonces } x(k+1) = A_i x(k) + B_i u(k) \quad (38)$$

- donde:

- $x(k) = \{x_1(k), \dots, x_n(k)\}^T$,
- $u(k) = \{u_1(k), \dots, u_{N_i}(k)\}^T$
- y $i = 1, \dots, L$ siendo L el número de reglas.

Síntesis directa

Gain Scheduling (Planificación de la ganancia)

- El modelo dinámico viene definido por el sistema borroso viene expresado por:

$$x(k+1) = \frac{\sum_{i=1}^L \mu_i(x_k) [A_i x(k) + B_i u(k)]}{\sum_{i=1}^L \mu_i(x_k)} \quad (39)$$

- Fin L4